

Solution

25 June 2021 20:02

$$X = \begin{bmatrix} 1, 2, \dots, d \\ \vdots \\ n \end{bmatrix}_{n \times d}$$

(Column standardized)

$$\text{Solution to PCA}$$

$$S = \text{cov}(X) = \frac{1}{n-1} \begin{bmatrix} 1, & 2, & \dots, & d \end{bmatrix}_{d \times d} = \boxed{\frac{1}{(n-1)} (X^T X)}$$

Because
mean = 0

eigen values of $\boxed{\text{cov}(X) = S}$ are, $\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_d$
 $v_1, v_2, v_3, \dots, v_d$

Corresponding eigen vectors

$$\boxed{S v_i = \lambda_i v_i}$$

$\underbrace{d \times d}_{d \times 1} \quad \underbrace{d \times 1}_{d \times 1}$

$$v_i \perp v_j \Rightarrow v_i^T v_j = 0$$

$u_j = v_j$ = eigen vector of S
corresponding to eigen value λ_j

Summary

$$X = \boxed{\quad}$$

- ① Column standardization of X
- ② $\text{cov}(X) = S = \frac{1}{(n-1)} X^T X$
- ③ Calculate eigen values & eigen vectors of S .
 $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \lambda_4 \geq \dots \geq \lambda_d$
 $v_1, v_2, v_3, v_4, \dots, v_d$

$$\textcircled{4} \quad u_1 = v_1, u_2 = v_2, \dots$$

choose first d' eigen vectors to project
d-dim data to d' -data space. ($d' \ll d$)

Just dot product of eigen vector with X , w^{*1}
transform d-dim to d' -dim

Interpretation of eigen values: —

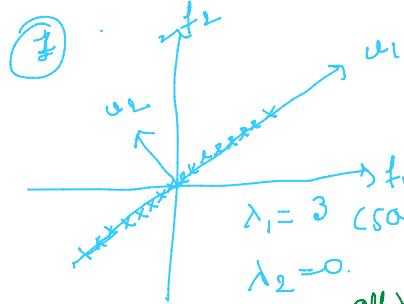
~ ~ ~ will have explained variance

Interpretation of eigen values:-

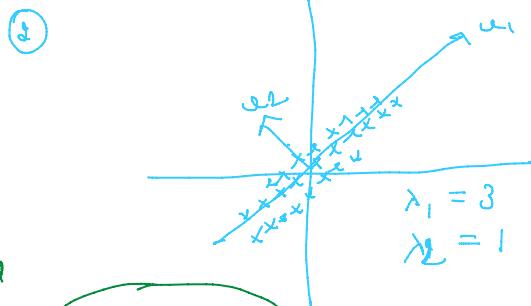
Eigen values will tell about explained variance.

$$\boxed{\frac{\lambda_i}{\sum \lambda_j}}$$

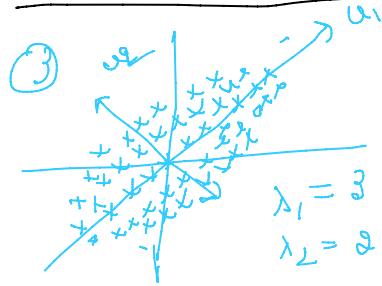
This will give % of variance explained/preserved by eigen vector corresponding to λ_i .



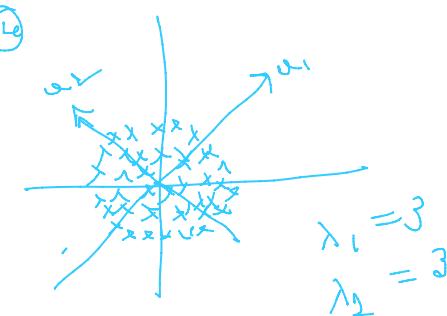
Here, $\frac{\lambda_1}{\lambda_1 + \lambda_2} = \frac{3}{3} = 100\%$ all variance preserved along u_1



$\frac{\lambda_1}{\lambda_1 + \lambda_2} = \frac{3}{4}$ 75% variance preserved along u_1 & remaining along u_2



Then, $\frac{\lambda_1}{\lambda_1 + \lambda_2} = \frac{3}{5} = 60\%$
60% of variance explained by u_1



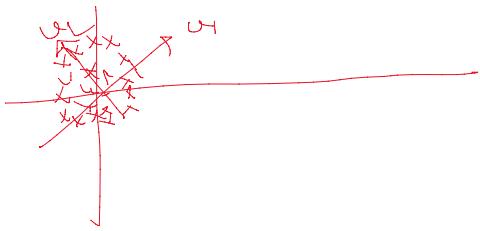
Then, $\frac{\lambda_1}{\lambda_1 + \lambda_2} = \frac{\lambda_2}{\lambda_1 + \lambda_2} = \frac{1}{2} = 50\%$

u_1 preserves 50% of variability
 u_2 also preserves 50% of variability

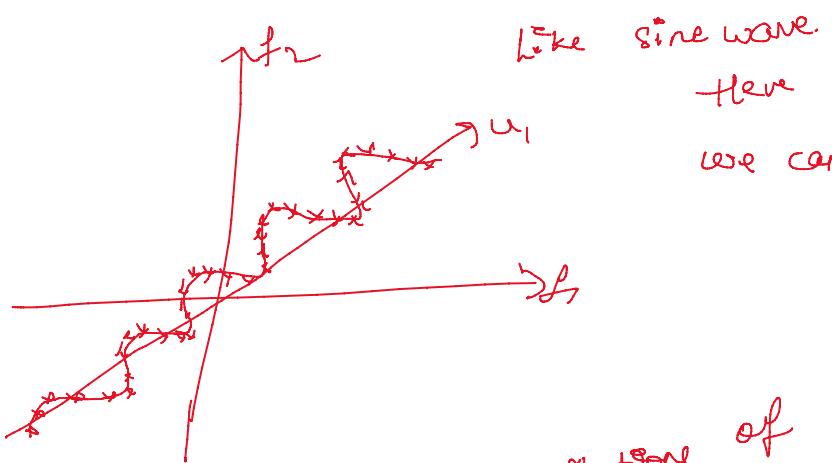
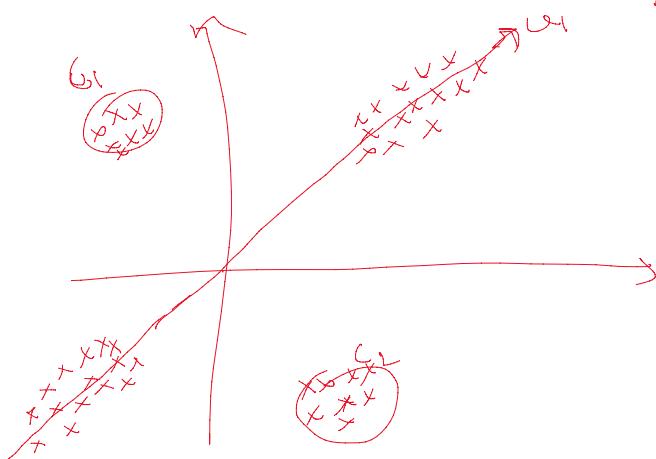
LIMITATIONS :-

Data is spherically distributed then $\lambda_i \sim \lambda_j$,
so more info will be lost if we do PCA.





Here when we perform PCA,
 C_1 & C_2 will be projected at origin
 then we can't distinguish C_1 & C_2
 then PCA fails



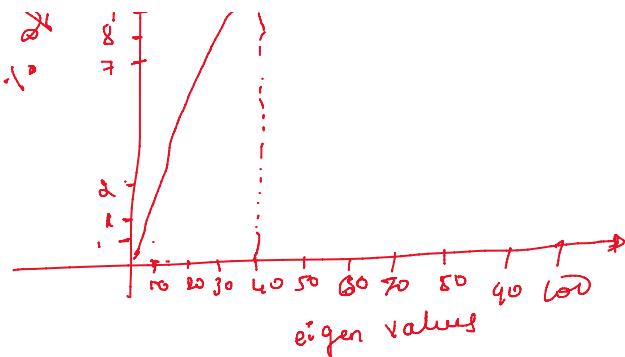
like sine wave.
 Here we can find u_1 , but
 we can't preserve the shape of sine wave.

These are all limitations of PCA

We use PCA to reduce dimension from d to d' ($d' \ll d$)
 while reducing based on required % of preservation of variance,
 we can choose value of d' .



⇒ there mostly 95% variance is explained
 by 10 features



→ by 40 features
rest 60 feature contributing only
5% variance.

Hence we can reduce $d(100)$ to $d'(40)$.

==>
this plot will help so much in real world.

This is all about PCA

Scikit-learn has direct function to perform PCA.

We can implement PCA without skLearn.

Simple code illustration is attached